



Jakarta Struts: Declarative Exception Handling Struts 1.2 Version

Core Servlets & JSP book: www.coreservlets.com

More Servlets & JSP book: www.moreservlets.com

Servlet, JSP, Struts, JSF, and Java Training Courses:
courses.coreservlets.com

Slides © Marty Hall, <http://www.coreservlets.com>, books © Sun Microsystems Press



For live Struts training, please see
JSP/servlet/Struts/JSF training courses at
<http://courses.coreservlets.com/>.



Taught by the author of *Core Servlets and JSP*, *More Servlets and JSP*, and this tutorial. Available at public venues, or customized versions can be held on-site at your organization.

Slides © Marty Hall, <http://www.coreservlets.com>, books © Sun Microsystems Press

Agenda

- **Transferring to custom error pages when various exception types are thrown**
 - Global exceptions
 - Per-Action exceptions
 - Standard web.xml approach
- **Using custom exception handlers**
 - Declaring
 - Defining

Idea

- **Programmatic exception handling**
 - Explicit try/catch blocks in any code that can throw exception
 - Works well when custom *value* (i.e., of variable) needed when error occurs
- **Declarative exception handling**
 - Entries in struts-config.xml (or web.xml) indicate what pages should be displayed when various errors occur

```
<exception key="some.key"
           type="package.SomeException"
           path="/WEB-INF/somepage.jsp"/>
```
 - Works well when custom *page* needed when error occurs
 - The Struts approach applies *only* to exceptions thrown by Actions

Entries in struts-config.xml

- **exception element with 4 possible attributes**
 - **key**: entry in a resource bundle (properties file) used to build an ActionMessage and that can be output with html:errors. Ignored now: see validation section
 - **type**: the fully qualified class of the exception. The most specific match is used. Applies *only* to Actions.
 - **path**: the address (relative to the Web app) of the error page. Should normally be in WEB-INF.
 - **handler**: the class that will handle the error. Usually omitted, and default handler used.
- **Used within global-exceptions or action**

```
<exception key="errors.null"
           type="java.lang.NullPointerException"
           path="/WEB-INF/errors/null.jsp"/>
```

6

Apache Struts: Declarative Exception Handling

www.coreservlets.com

Three Basic Approaches

- **Give page for specific type of exception**
 - Use **exception** entry in **global-exceptions** or **action**
- **Give page for generic Exception**
 - Use **exception** entry in **global-exceptions**, or
 - Use **exception-type** entry in web.xml, or use **error-page** attribute of **page** directive
 - Advantage of page directive: the **exception** object is bound, so you can print a backtrace
- **Give custom handler for specific type of exception**
 - Use **exception** entry in **global-exceptions** or **action**, and
 - Define subclass of **ExceptionHandler**

7

Apache Struts: Declarative Exception Handling

www.coreservlets.com



Handling Specific Exceptions

Slides © Marty Hall, <http://www.coreservlets.com>, books © Sun Microsystems Press

Example 1: Page for NullPointerException

- **struts-config.xml**

```
<global-exceptions>
  <exception key="some.key"
             type="java.lang.NullPointerException"
             path="/WEB-INF/errors/null.jsp"/>
</global-exceptions>
```
- **null.jsp**

```
<!DOCTYPE ...>
<HTML>
<HEAD><TITLE>NullPointerException</TITLE></HEAD>
<BODY BGCOLOR="#FDF5E6">
<H1>NullPointerException</H1>
Blah, blah, blah, blah.
</BODY></HTML>
```

Example 1: Page for NullPointerException (Bean)

```
package coreservlets;

import org.apache.struts.action.*;

public class ColorBean extends ActionForm {
    private String foregroundColor, backgroundColor;

    public String getForegroundColor() {
        return(foregroundColor);
    }

    public void setForegroundColor(String fgColor) {
        if (!fgColor.equals("")) {
            foregroundColor = fgColor;
        }
    }

    ...
}
```

10

Apache Struts: Declarative Exception Handling

www.coreservlets.com

Example 1: Page for NullPointerException (Action)

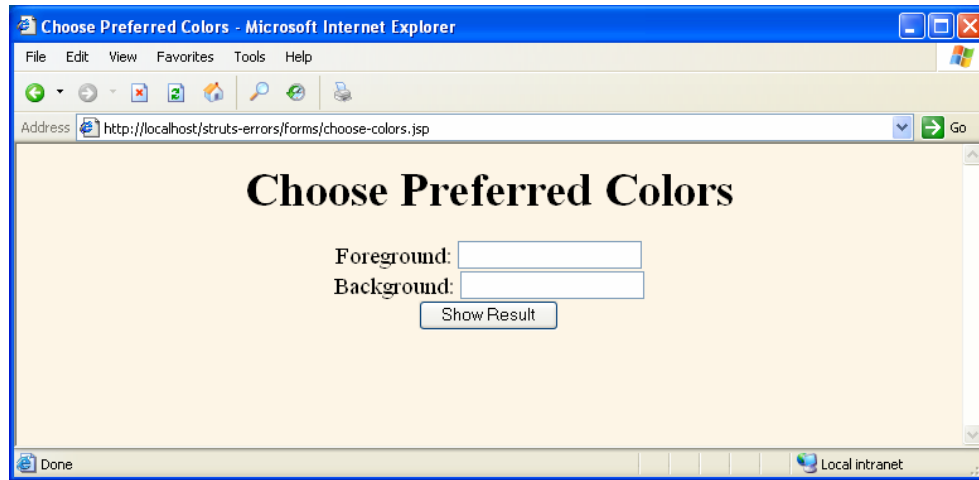
```
public class ShowColorsAction extends Action {
    public ActionForward execute(ActionMapping mapping,
                                ActionForm form,
                                HttpServletRequest request,
                                HttpServletResponse response)
        throws Exception {
        ColorBean colorBean = (ColorBean)form;
        String fg = colorBean.getForegroundColor();
        String bg = colorBean.getBackgroundColor();
        if (fg.equals(bg)) {
            colorBean.setForegroundColor("BLACK");
            colorBean.setBackgroundColor("WHITE");
        }
        ...
        return(mapping.findForward("success"));
    }
}
```

11

Apache Struts: Declarative Exception Handling

www.coreservlets.com

Example 1: Page for NullPointerException (Results)

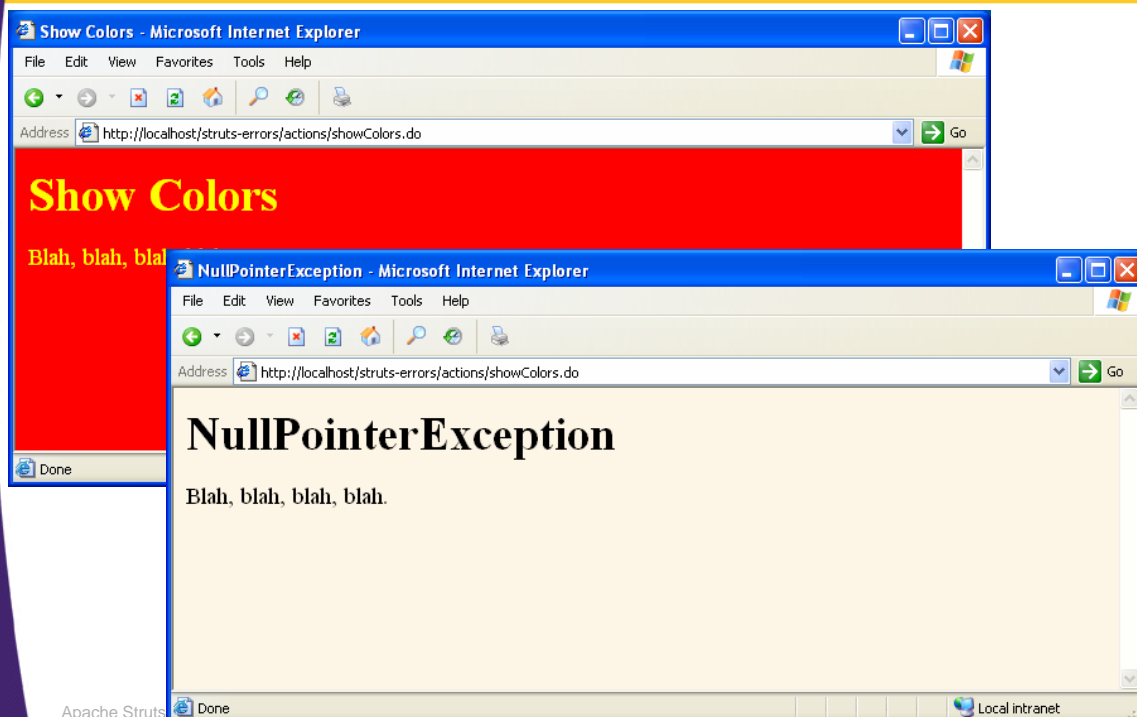


12

Apache Struts: Declarative Exception Handling

www.coreservlets.com

Example 1: Page for NullPointerException (Results)



13

Apache Struts

Done

Local intranet



Handling Generic Exceptions (Struts-Specific Approach)

Slides © Marty Hall, <http://www.coreservlets.com>, books © Sun Microsystems Press

Example 2: Page for Generic Exception

- **struts-config.xml**

```
<global-exceptions>
  <exception key="some.key"
    type="java.lang.NullPointerException"
    path="/WEB-INF/errors/null.jsp"/>
  <exception key="some.key"
    type="java.lang.Exception"
    path="/WEB-INF/errors/error.jsp"/>
</global-exceptions>
```
- **null.jsp**

```
<!DOCTYPE ...>
<HTML>
<HEAD><TITLE>Error</TITLE></HEAD>
<BODY BGCOLOR="#FDF5E6">
<H1>Error</H1>
Blah, blah, blah, blah.
</BODY></HTML>
```

Example 2: Page for Generic Exception (Action)

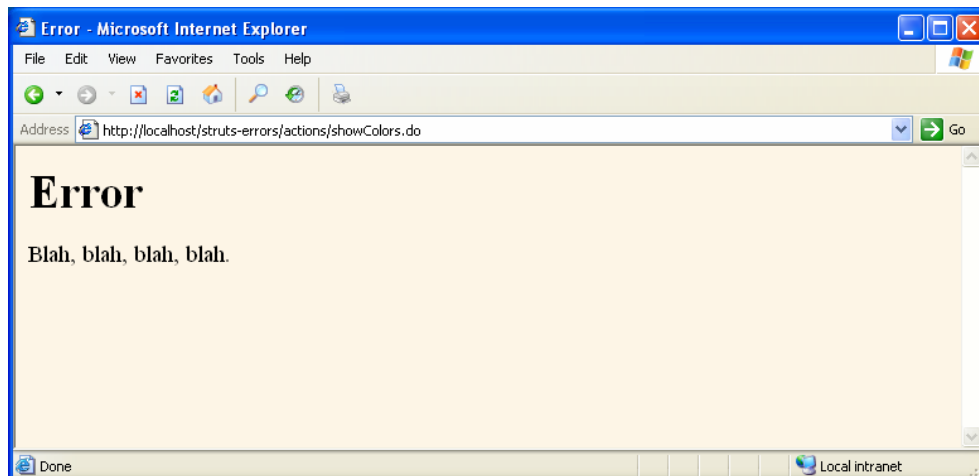
```
public class ShowColorsAction extends Action {
    public ActionForward execute(ActionMapping mapping,
                                ActionForm form,
                                HttpServletRequest request,
                                HttpServletResponse response)
        throws Exception {
        ColorBean colorBean = (ColorBean)form;
        String fg = colorBean.getForegroundColor();
        String bg = colorBean.getBackgroundColor();
        if (fg.equals(bg)) {
            colorBean.setForegroundColor("BLACK");
            colorBean.setBackgroundColor("WHITE");
        }
        if (bg.equalsIgnoreCase("papayawhip")) {
            throw(new Exception("Bogus background color!"));
        }
        ...
        return(mapping.findForward("success"));
    }
}
```

16

Apache Struts: Declarative Exception Handling

www.coreservlets.com

Example 2: Page for Generic Exception (Result)



17

Apache Struts: Declarative Exception Handling

www.coreservlets.com



Handling Generic Exceptions (Standard Approaches)

Slides © Marty Hall, <http://www.coreservlets.com>, books © Sun Microsystems Press

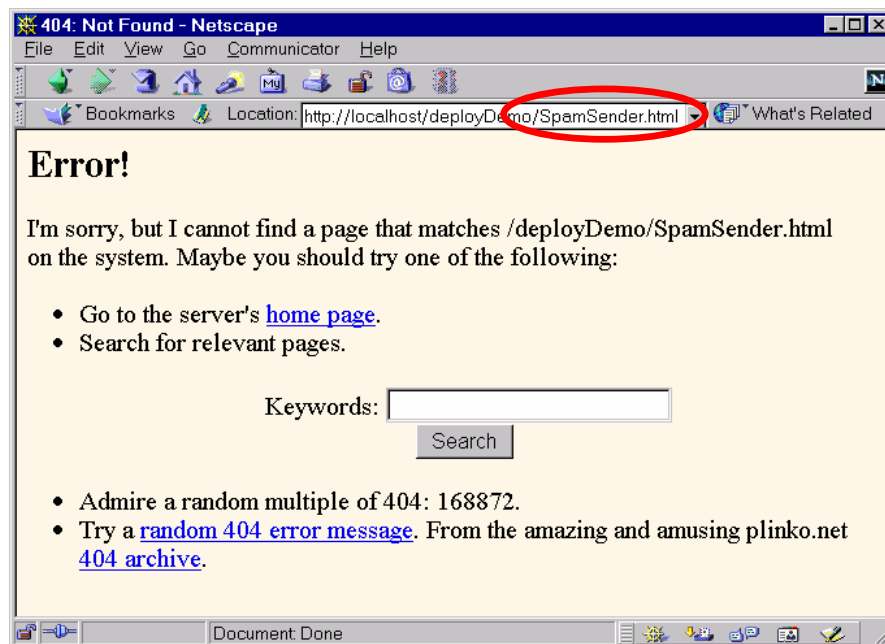
Alternatives Using Standard Servlet/JSP Elements

- **Pages to use for specific HTTP status codes**
 - Use the `error-code` element
 - Within `error-page`
- **Pages to use when specific uncaught exceptions are thrown**
 - Use the `exception-type` element
 - Within `error-page`
- **Page-specific error pages**
 - Use `<%@ page errorPage="Relative URL" %>`
 - In individual JSP page, not in `web.xml`

Error Pages and Status Codes

```
<web-app...>  
  <error-page>  
    <error-code>404</error-code>  
    <location>/WEB-INF/NotFound.jsp</location>  
  </error-page>  
  ...  
</web-app>
```

Error Pages and Status Codes: Result



Error Pages and Exceptions: A Dangerous Computation

```
package moreservlets;

/** Exception used to flag particularly onerous
    programmer blunders. Used to illustrate the
    exception-type web.xml element.
 */

public class DumbDeveloperException extends Exception {
    public DumbDeveloperException() {
        super("Duh. What was I *thinking*?");
    }

    public static int dangerousComputation(int n)
        throws DumbDeveloperException {
        if (n < 5) {
            return(n + 10);
        } else {
            throw(new DumbDeveloperException());
        }
    }
}
```

22

Apache Struts: Declarative Exception Handling

www.coreservlets.com

Error Pages and Exceptions: A Risky Page

```
<!DOCTYPE HTML PUBLIC
    "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD><TITLE>Risky JSP Page</TITLE></HEAD>
<BODY BGCOLOR="#FDF5E6">
<H2>Risky Calculations</H2>
<%@ page import="moreservlets.*" %>
<% int n = ((int)(10 * Math.random())); %>
<UL>
    <LI>n: <%= n %>
    <LI>dangerousComputation(n):
        <%= DumbDeveloperException.dangerousComputation(n) %>
</UL>
</BODY></HTML>
```

23

Apache Struts: Declarative Exception Handling

www.coreservlets.com

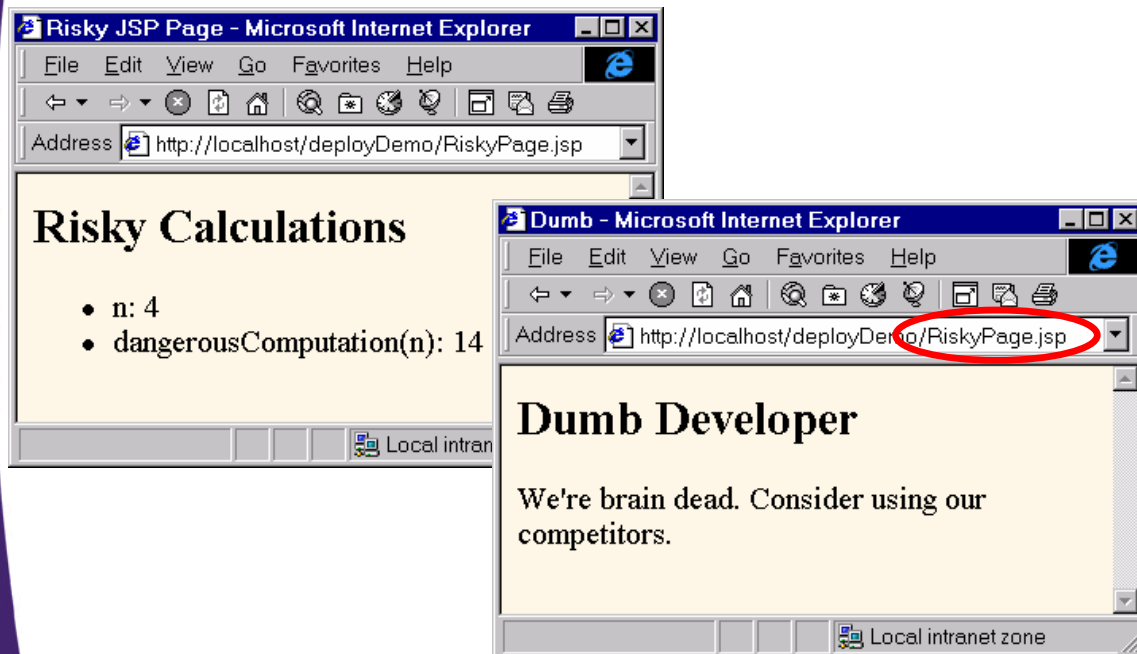
Error Pages and Exceptions: Declaring Error Page for DDE

```
<web-app...>
  <error-page>
    <exception-type>
      moreservlets.DumbDeveloperException
    </exception-type>
    <location>/WEB-INF/DDE.jsp</location>
  </error-page>
  ...
</web-app>
```

Error Pages and Exceptions: WEB-INF/DDE.jsp

```
<!DOCTYPE HTML PUBLIC
    "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD><TITLE>Dumb</TITLE></HEAD>
<BODY BGCOLOR="#FDF5E6">
<H2>Dumb Developer</H2>
We're brain dead. Consider using our competitors.
</BODY></HTML>
```

Error Pages and Exceptions: Results



26

Apache Struts: Declarative Exception Handling

www.coreservlets.com

Page-Specific Error Pages: Example

```
...<BODY>
<%@ page errorPage="/WEB-INF/SpeedErrors.jsp" %>
<TABLE BORDER=5 ALIGN="CENTER">
  <TR><TH CLASS="TITLE">Computing Speed</TH></TR></TABLE>
<%!
private double toDouble(String value) {
    return(Double.parseDouble(value));
}
%>
<%
double furlongs = toDouble(request.getParameter("furlongs"));
double fortnights =
    toDouble(request.getParameter("fortnights"));
double speed = furlongs/fortnights;
%>
<UL>
  <LI>Distance: <%= furlongs %> furlongs.
  <LI>Time: <%= fortnights %> fortnights.
  <LI>Speed: <%= speed %> furlongs per fortnight.
</UL>
</BODY></HTML>
```

27

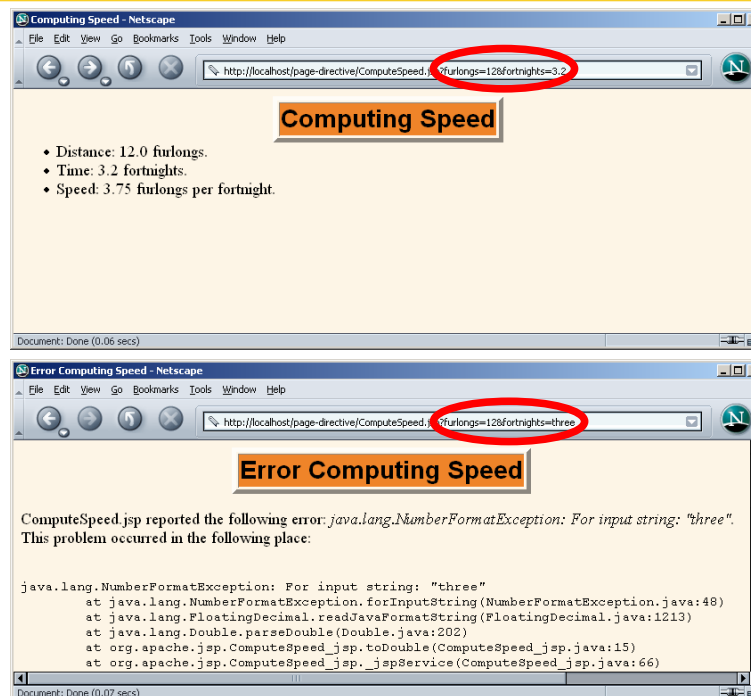
Apache Struts: Declarative Exception Handling

www.coreservlets.com

Page-Specific Error Pages: Example (Continued)

```
...<BODY>
<%@ page isErrorPage="true" %>
<TABLE BORDER=5 ALIGN="CENTER">
  <TR><TH CLASS="TITLE">
    Error Computing Speed</TH></TR></TABLE>
<P>
ComputeSpeed.jsp reported the following error:
<I><%= exception %></I>. This problem occurred in the
following place:
<PRE>
<%@ page import="java.io.*" %>
<% exception.printStackTrace(new PrintWriter(out)); %>
</PRE>
</BODY></HTML>
```

Page-Specific Error Pages: Results





Custom Exception Handlers

Slides © Marty Hall, <http://www.coreservlets.com>, books © Sun Microsystems Press

Using Custom Handlers

- **Extend ExceptionHandler**
 - Use execute method, but with two extra arguments (Exception and ExceptionConfig)
 - Give custom behavior (recording error in database, sending email, etc.)
 - Return super.execute in most cases
- **Declare handler in struts-config.xml**

```
<exception key="some.key"  
           type="pkg.SomeException"  
           handler="pkg.SomeExceptionHandler"  
           path="/WEB-INF/somefile.jsp"/>
```

Example 3: Custom Handler (struts-config.xml)

```
<global-exceptions>
  <exception key="some.key"
    type="java.lang.NullPointerException"
    path="/WEB-INF/errors/null.jsp"/>
  <exception key="some.key"
    type="coreservlets.CriticalProblem"
    handler=
      "coreservlets.CustomExceptionHandler"
    path="/WEB-INF/errors/error.jsp"/>
  <exception key="some.key"
    type="java.lang.Exception"
    path="/WEB-INF/errors/error.jsp"/>
</global-exceptions>
```

Example 3: Custom Handler (Custom Handler)

```
package coreservlets;

import javax.servlet.http.*;
import org.apache.struts.action.*;
import org.apache.struts.config.*;
import javax.mail.*;
import javax.mail.internet.*;
import java.util.*;
import javax.servlet.*;

public class CustomExceptionHandler extends ExceptionHandler {
  public ActionForward execute(Exception exception,
    ExceptionConfig config,
    ActionMapping mapping,
    ActionForm form,
    HttpServletRequest request,
    HttpServletResponse response)
    throws ServletException {
```

Example 3: Custom Handler (Custom Handler, Continued)

```
try {
    Properties properties = new Properties();
    properties.put("mail.host", "coreservlets.com");
    Session mailSession =
        Session.getDefaultInstance(properties, null);
    MimeMessage mailMessage = new MimeMessage(mailSession);
    mailMessage.setFrom
        (new InternetAddress("errors@some-company.com"));
    mailMessage.setFrom
        (new InternetAddress("hall@coreservlets.com"));
    mailMessage.setSubject("Error in Struts application");
    mailMessage.setText
        ("The struts-errors app had a critical problem:\n" +
         " " + exception + "\n\n" +
         "Get your butt out to the office and fix it NOW.");
    Transport.send(mailMessage);
    System.out.println("Sent emergency email.");
} catch(Exception e) {}
return(super.execute(exception, config, mapping,
                    form, request, response));
}}
```

Example 3: Custom Handler (Custom Exception Class)

```
package coreservlets;

public class CriticalProblem extends Exception {
    public CriticalProblem(String message) {
        super(message);
    }
}
```

Example 3: Custom Handler (Action)

```
public class ShowColorsAction extends Action {
    public ActionForward execute(ActionMapping mapping,
                                ActionForm form,
                                HttpServletRequest request,
                                HttpServletResponse response)
        throws Exception {
        ColorBean colorBean = (ColorBean)form;
        String fg = colorBean.getForegroundColor();
        String bg = colorBean.getBackgroundColor();
        if (fg.equals(bg)) {
            colorBean.setForegroundColor("BLACK");
            colorBean.setBackgroundColor("WHITE");
        }
        if (bg.equalsIgnoreCase("papayawhip")) {
            throw(new Exception("Bogus background color!"));
        }
        if (Math.random() < 0.1) {
            String message =
                "Database server exploded. Building is burning!";
            throw(new CriticalProblem(message));
        }
        return(mapping.findForward("success"));
    }
}
```

36

Apache Struts: Declarative Exception Handling

www.coreservlets.com

Summary

- **Give page for specific type of exception**
 - Use **exception** entry in **global-exceptions** or **action**
- **Give page for generic Exception**
 - Use **exception** entry in **global-exceptions**, or
 - Use **exception-type** entry in **web.xml**, or use **error-page** attribute of **page** directive
- **Give custom handler for specific type of exception**
 - Use **exception** entry in **global-exceptions** or **action**, and
 - Define subclass of **ExceptionHandler**

37

Apache Struts: Declarative Exception Handling

www.coreservlets.com



Questions?

Core Servlets & JSP book: www.coreservlets.com
More Servlets & JSP book: www.moreservlets.com
Servlet, JSP, Struts, JSF, and Java Training Courses:
courses.coreservlets.com
Slides © Marty Hall, <http://www.coreservlets.com>, books © Sun Microsystems Press